Automatic Analysis of Plot for Story Rewriting

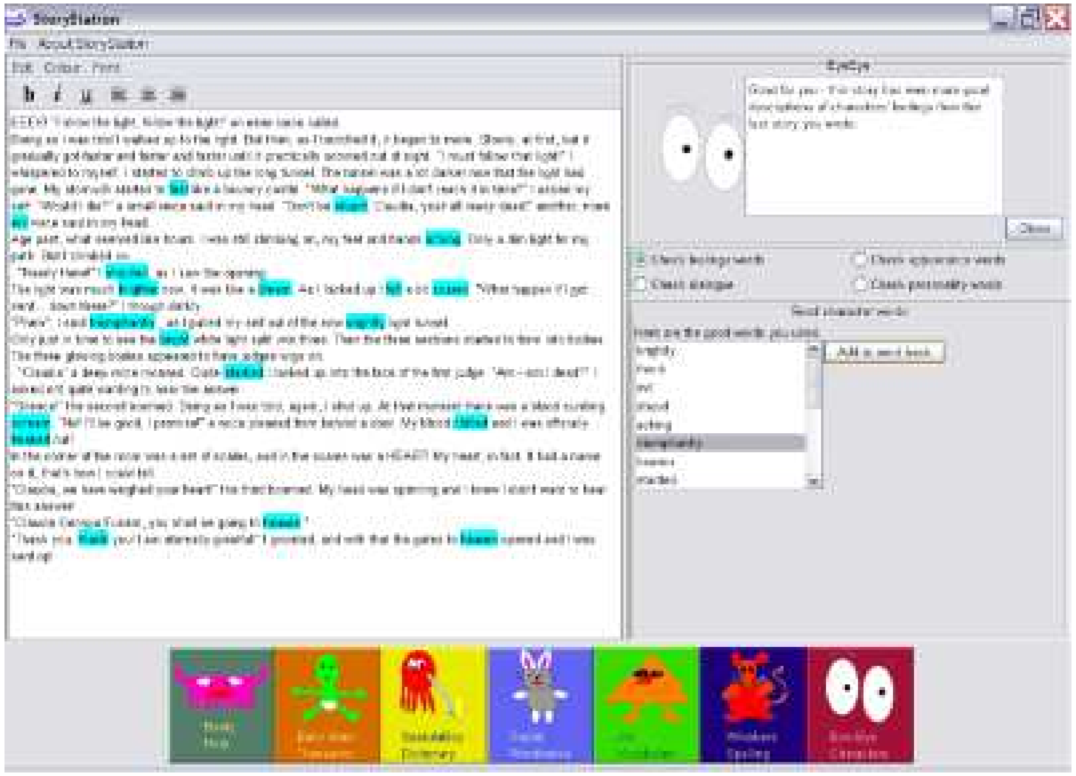
Harry Halpin, Johanna D. Moore, and Judy Robertson

HCRC/ICCS University of Edinburgh

School of informatics

StoryStation

StoryStation: Activating *Pinky the Plot Analyzer* in StoryStation.



Story Rewriting Task

- **Story Rewriting**: Students are read a story (the *exemplar story*) and rewrite it (the *rewritten story*), allowing them to focus on skills such as diction without the cognitive load of plot creation.

- **Plot Analysis**: Compare an exemplar story to a rewritten story for similarity. Is the rewritten story coherent? Did the pupil remember the events?

- Not used for automated grading, instead enables agent to provide help to student or to encourage student to ask for help from the teacher.

Corpus Creation

Size: 103 rewritten stories from one exemplar story from two classes.

Rating Scale: Teacher devised 4-tier ranking scheme based on ability of the student to recall of events and "understand the point."

Excellent	Understands "point" and recalls "right details."
Good	Recalls events but no "deeper understanding of plot."
Fair	Lacks more than one "chunk" of the story.
Poor	Lacks "substantial" part of plot.

Raters

- Stories graded by 3 independent raters.
- Absolute agreement of raters low (Avg. raw agreement among raters 58%).
- Yet disagreement was **highly** systematic.
- The two raters with the highest disagreement (39%) had Cronbach's α of .86 and Kendall's τ_b of .72.

Overview of Plot Analysis

For each rewritten story:

- Automatically identify events.
- Use Plot Comparison Algorithm to compare events with exemplar story's events.
- Perform LSA comparison with exemplar story text.
- Use results of above as features in machine-learner to rate stories automatically.

Event Calculus

Motivation: Need to automatically identify "events" in students' stories.

Uses a simplified version of the **event calculus** since it captures relevant aspects of plot, such as *temporal order*, *characters*, and *events*.

Automatically converts raw text of story to a sequence of events via an XML-based **pipeline** using POS tagging, anaphora resolution, and chunking to extract event calculus representations from often ungrammatical text.

Sentence	**Event**
Nils stays in Sweden	*stay(t=1, Nils, Sweden)*
And he is always playing with geeseon his mountain.	*play(t=1,Nils,geese, mountain)*
He sees a bird	*see(t=2, Nils, bird)*

Plot Comparison Algorithm

Motivation: Need a measure of difference between the events in the rewritten story and exemplar story.

The algorithm iterates through the exemplar story's events looking for matches in each event of the rewritten story.

Events are matched by matching their components one at a time, starting with the event name and then matching entities.

Uses WordNet to find out if synonyms were used and uses a "now-point" to determine if the event is in the correct temporal order.

Example

Exemplar	Rewritten
throw(t=1,Nils, coin)	*toss(t=3,coin)*

Algorithm Result: (synonym, out of order, no match, exact match)

The algorithm produces a feature vector of these results for each rewritten story by encoding these results as integers.

Classifying Stories

Machine-learning allows the significance of the presence or absence of particular events to be found. Features include similarity discovered with word frequency comparison in a reduced subspace (LSA) and the results of the Plot Comparison Algorithm (Events).

Results using 10-fold cross-validation with only a few representative results:

Machine Learner	**Features**	**% Correct**
ID3 Decision-Tree	Events	40.66%
K-Nearest Neighbors	LSA	44.66%
Naive Bayes	LSA and Events	**54.37%**
Avg. Rater Agreement		58.37%

Using event calculus-based features significantly helps increase performance of overall system.

Sample Results

This confusion matrix shows the distribution of ratings output by *Naive Bayes*.

Class	1	2	3	4
1 (Excellent)	0	**17**	1	0
2 (Good)	1	**29**	2	1
3 (Fair)	0	13	5	1
4 (Poor)	0	8	3	**22**

- Naive Bayes collapses all categories except "good" and "poor," but has a low error spread for these ratings.

Discussion: Future Work

- The rating scale is fairly subjective.
- Need a larger corpus.
- Use additional linguistic features.
- More complex temporal order extraction.
- The Plot Comparison Algorithm may be genre-dependent (narratives).

Discussion: Advantages

Use of events allows agent to give feedback to student about what events are missing, characters misidentified, and such.

Why does using deep features help? Perhaps using “deep” features bring out relevant features not obvious in shallow features, bringing to prominence features that otherwise might be lost, especially in small data-sets.

It’s hard to tell what level of abstraction is correct for a given task and corpus, but preliminary results are encouraging for this task.

Automated plot analysis of the story rewriting task solves a real-world problem using a mixture of shallow and deep features.